Univers
USB

The Univers*USB solution

Gianluca Moro

Department of Statistical Sciences

University of Padova

Gianluca.Moro@unipd.it

Revision 20140117

Contents

1 Introduction **5**
- 1.1 What is it? . . . 5
- 1.2 Who uses it? . . . 5
- 1.3 From where does it come from? . . . 5
- 1.4 What does it mean? . . . 6

2 User Manual **7**
- 2.1 System requirements . . . 7
- 2.2 Howto use it . . . 7
- 2.3 A less short explanation: . . . 8
- 2.4 First boot . . . 8
- 2.5 How to put UniversUSB on USB flash drive . . . 10

3 Developer Manual: build your own UniversUSB **11**
- 3.1 Plain installation . . . 11
- 3.2 The building script . . . 12
  - 3.2.1 Comments . . . 12
  - 3.2.2 Variables . . . 12
  - 3.2.3 Functions . . . 13
  - 3.2.4 Package personalization . . . 13
  - 3.2.5 String parsing . . . 14
  - 3.2.6 The repository . . . 14
  - 3.2.7 Package installation . . . 14
  - 3.2.8 Background and Firefox home page . . . 15
  - 3.2.9 PAM . . . 15
  - 3.2.10 DNS setup . . . 16
  - 3.2.11 Remastersys . . . 16
  - 3.2.12 The Setup.sh script . . . 17

4 References **19**

Chapter 1

Introduction

1.1 What is it?

UniversUSB, http://www.universusb.org/, is a portable Linux OS installed on a USB stick, with a set of predefined but personalizable software. You keep your data and your programs in your pocket; use them on every computer with USB boot enabled, without worrying about letting around some unwanted logs and this without giving away your data security or privacy.

After you have your preferred software and configurations installed and configured, you can have your personal working environment everywhere you find a USB bootable PC.

1.2 Who uses it?

The solution can be used by students who need a portable solution, with specific software tools: the initial version contains statistical and mathematical tools, but other version can be easily personalized with the building script, from having just the tools to browse internet, to having some complete set of network troubleshooting tools.

1.3 From where does it come from?

The project derives from CloudUSB (now dismissed) and is developed at Department of Statistical Sciences, University of Padova. This is the reason of the initial selection of default software, but a personalization can be as easy as inserting in the building script a list of desired software.

1.4 What does it mean?

The meaning of the Univers*USB can be:

- the UNIVERSe USB
- the UNIVERSity USB solution
- the UNIVERSal USB solution

as you prefer: and considering the ancestor CloudUSB project, from which this derives, we can say that we evolved **from Cloud to Universe!**

Chapter 2

User Manual

2.1 System requirements

UniverUSB is based on Ubuntu, http://www.ubuntu.com/ so the requirements are the same.

- To run UniversUSB:
  - a usb flash drive: recommended 8G or more
  - the host PC must be able to boot from USB and run Ubuntu Linux: see Ubuntu System Requirements
- To create your own UniversUSB, same as before, plus:
  - UniversUSB ISO
  - a host PC with a Linux OS and the UnetBootIn program installed

2.2 Howto use it

To make it short:

- connect your UniversUSB to a PC
- boot it from USB and use your system as a normal Linux system
- put in the **data** folder all your NON reserved data
- in the **private-data** folder all your reserved data

That's all!
The default login username and password are both "universusb"

2.3 A less short explanation:

The configured UniversUSB is very easy to use; it comes as a Linux system on USB stick, configured in persistent mode: it behaves as a standard hard-disk installation which means that you can write documents, modify settings and they will be saved on the flash drive for the next time you use it, as in your desktop PC!

After you have configured your Dropbox account and your private data folder you will get 2 folders on your desktop: a **data** folder and a **private-data**. They are both syncronized by Dropbox on their online server, so if you lose your USB flash drive, you have an on-line backup, or you can work on the same data both from UniversUSB flash drive and from your desktop computer (if this is connected to the same Dropbox account)

You could have some perplexities to send your data on a remote server, via Internet, but in this case you just put your sensitive data in the private-data folder: this folder is protected by a password, the data you put in is crypted locally, and sent over the Internet to the Dropbox storage **in crypted form**.

The data in the private-folder is stored in a crypted format even locally in the USB flash drive, and made available only if you login in the system with the correct password.

Note that if you lose the usb flash drive, and someone access your data, he will NOT see your private data!

2.4 First boot

The UniversUSB needs to be configured before you use it. The steps you need to accomplish are the following:

- Boot the UniversUSB flash drive: the first time the boot will be somewhat slower because it has to initialize the persistent data partition
- After the boot, change the password! **The default login username and password are both "universusb"**. You need to change the password to keep your privacy! The password you use to login will be the same used to crypt your sensitive data, so choose a good password!

  1. go to System Settings
  2. select User Accounts
  3. click on the field of the password (the one with big black points)
  4. write the old (universusb) password and set a new one

5. restart the computer

- Run the script **Setup** whose icon is on the Desktop; this will download the Dropbox client (you need to be connected to internet to accomplish this step!) and install it. If you have an account, use it, otherwise create a new account. In details:

  1. ask confirm to install Dropbox: say Ok. Dropbox client is downloaded and run
  2. select the relevant option: if you do not have a Dropbox account, you can create one, otherwise use your account
  3. select Typical installation
  4. click sometimes on Next
  5. now the private-data directory is going to be configured for crypting: say Yes (see the consideration in the next paragraph)
  6. insert your login password
  7. all done: now restart

At step 5, if you never used that Dropbox account with UniversUSB private data, when you are asked **Do you want to initialize the private-data directory? (Only needed on new Dropbox account)** answer **OK**: this will create the crypted folder to store your data. **Set the same password as the login one!** If you want to use UniversusUSB with a preexistent account already configured to use EncFS, just click the No button and you are done.

NOTE ON PASSWORD: The password for login must be the same as the one used for EncFS, so, if you have an already configured EncFS directory, just set the login password equal to the EncFS password.

It is possible to change the EncFS password with:

```
encfsctl passwd /home/universusb/Dropbox/private-data
```

If the passwords are not the same, and for some reason you want to keep them different, then you just need to manually mount the crypted partition:

```
encfs /home/universusb/Dropbox/private-data
  /home/universusb/Desktop/private-data
```

2.5 How to put UniversUSB on USB flash drive

UniversUSB is made available as ISO image: to store on a USB flash drive you need the Unetbootin tool, http://unetbootin.sourceforge.net/, and the ISO image file. The steps are:

1. select DiskImage and choose the UniversUSB ISO filename
2. set Space used to preserve file across reboot: it is the space you will have available for your personal data/configuration: try to set to 1000 Mb: remember that the base system is 4G, and the all must fit in the physical USB flash drive size.
3. type USB drive ad destination, and set Drive to the drive where the USB flash drive is connected: usually the default is correct.

   WARNING: if unsure, double check the destination!
4. just press OK: now it will need some time to copy all the needed data.

Chapter 3

Developer Manual: build your own UniversUSB

The user can build his own UniversUSB to personalize it for some specific task, to fit it on a smaller device, or just to learn how to do it. The process has essentially 3 main steps:

1. install a plain Ubuntu Linux
2. personalize with the official building script or with your version of it
3. put the result on a USB stick

3.1 Plain installation

The UniversUSB is based on a Ubuntu (13.10 at the moment) distribution. The initial choice can be different, but the building script uses the Ubuntu conventions about configuration files, so if you change distribution, probably there will be some work to adapt the scripts.

The same Ubuntu versions have more or less minor changes from version to version, so, with upcoming releases there could be the necessity to change some script lines ...

After having selected a distribution, you just need to install it as you would normally do, more or less in this way:

- Language English (or your preferred one)
- Install Ubuntu
- English (again - the one you use)

- Check Download updates while installing
- Check Install this third-party software
- Erase disk and install Ubuntu
- Set CET time (Rome) (according to your position)
- Set keyboard english (or the layout you prefer)
- User Name UniversUSB, pc name universusb
- new user login: universusb, password universusb
- Ubuntu account: "Log in later"

You can personalize the username, but the default building script requests that the working user is `universusb`: if you change it, change accordingly the building script.

3.2 The building script

The creation of the UniversUSB starts from a plain Ubuntu installation, as we have seen, and is personalized with a script: in this chapter we see parts of the script, which can be useful to see some bash programming ideas both to personalize a distribution, or other bash scripting tasks.

3.2.1 Comments

Nothing to say: use the comments to explain what you do - it will be useful to people who read the script, but to the author too if in some years he will need to look at his code!

3.2.2 Variables

The script uses some variables: they are always useful to have a definition in just one place. In our script it is used to define some URL and filenames used to build the UniversUSB.

```
VERSION="0.1"
DEFAULT_USERNAME="universusb"
```

After the declaration, the usage is

```
echo $VERSION
```

3.2.3 Functions

Functions are very useful for repeated code: in this case we use it to print logging messages to debug the script: the line `read -p ...`, usually commented out, can be uncommented to pause the script and check what is going on. One uncomment will enable the pause all along the script.

```
function debug {
    echo -n "Debug: "
    echo $1
#    read -p "Press [Enter] key to continue ..."
}

debug "Make sure only root can run our script"
```

3.2.4 Package personalization

The script comes with some preinstalled packages: they are organized in groups:

```
INSTALL_LATEX=yes
#INSTALL_DEVEL=yes
INSTALL_MATH=yes
INSTALL_STAT=yes
```

A specific group of package is enabled if the corresponding variable is **uncommented**: to remove a set of packages, just **comment** the correspondinf line. Note that writing `INSTALL_MATH=no` will not remove the mathematical packages, you need to comment it out as in `#INSTALL_MATH=yes`.

This depends on how the variable is used:

```
if [ x$INSTALL_MATH != "x" ] ; then
...
apt-get install -y packagename

fi
```

Each set of packaget is included in a `if ... then ... fi` block, and the block is executed if the corresponding varialbe is defined, i.e. has some value - every value will make true the condition!

3.2.5 String parsing

The problem here is to find the name of the Desktop folder, which can change acordingly to localization.

```
#
# Look for "Desktop" name in localized systems
#
if [ -f /home/$DEFAULT_USERNAME/.config/user-dirs.dirs ] ;
then
  . /home/$DEFAULT_USERNAME/.config/user-dirs.dirs
  XDG_ONLY_DESKTOP_DIR=‘echo $XDG_DESKTOP_DIR |
    awk ’BEGIN { FS = "/" } ; { print $NF }’‘
  DESKTOP_DIR=/home/$DEFAULT_USERNAME/$XDG_ONLY_DESKTOP_DIR
else
  DESKTOP_DIR=/home/$DEFAULT_USERNAME/Desktop
fi

debug "Desktop directory is $DESKTOP_DIR"
```

The `awk` code `print $NF` prints the last fields of the vector, that is the name of the Desktop folder in localized system: in this way it works both if I run from root (`/root/Desktop`), or from normal user (`/home/user/Desktop`). The default option is to use `Desktop` string.

3.2.6 The repository

Ubuntu has some predefined repository to download software: we want to add some other sources for specific application. The solution is to find the official repository for the wanted application, and add it to the standard configuration file `/etc/apt/sources.list`. In our case we add repository for Remastersys, Dropbox, Skype. After modifying the `sources.list` file, the command `apt-get update` must be issued to update the indexes.

3.2.7 Package installation

The software personalization is done by installing the desired packages, for example:

```
apt-get install -y --force-yes dropbox
```

The option `-y` and `--force-yes` are used to force whenever possible a default YES answer to all the installation question. Usually this is a good option only if you know that the installation will go on smoothly.

3.2.8 Background and Firefox home page

The UniversUSB background can be personalized by the script with a `PNG` file: in this implementation it overwrite the default background which is stored in `/usr/share/backgrounds/warty-final-ubuntu.png`.

Firefox can be personalized with a web site home page by:

```
cat > /usr/lib/firefox/defaults/pref/home.js <<EOF
user_pref("browser.startup.homepage", "$DEFAULT_PROJECT_URL");
EOF
```

This code write everything included between `EOF` in the configuration file of the browser `/usr/lib/firefox/defaults/pref/home.js`

3.2.9 PAM

The core to have an encrypted remote copy of the data is given by:

- PAM
- encfs
- Dropbox
- mounting the private data encrypted file system

The idea is to have a `private-data` directory mounted by `encfs`, so it is crypted - for more information you can see http://www.arg0.net/encfs, and stored, in its crypted form both on the USB stick and on Dropbox. In this way the user is protected both is the stick is lost and on the online storage. As a matter of fact, Dropbox is just an option - other storage services can be used.

Note: the data is stored by EncFS: it is the user responsibility to store securely the password, otherwise all data is lost!

Note: the password user for the login must be the same used for the EncFS, otherwise the automatic mount will not work.

The PAM module is used to have an authomatic mounting of the encrypted partition. The relevant part of `/etc/pam.d/common-auth` is:

```
cat > /etc/pam.d/common-auth <<EOF
auth sufficient                  pam_encfs.so
auth [success=1 default=ignore] pam_unix.so nullok_secure use_first_pass
auth requisite                   pam_deny.so
auth required                    pam_permit.so
```

```
auth optional                   pam_mount.so
auth optional                   pam_ecryptfs.so unwrap
EOF
```

the file will declare that we use the `pam_encfs.so` module during the authentication, and in the `/etc/security/pam_encfs.conf` we configure what we want to do, essentialy the line

```
universusb  /home/universusb/Dropbox/private-data
  /home/universusb/Desktop/private-data -v allow_other,nonempty
```

which mounts the crypted directory stored in `Dropbox/private-data` to a local directory on the Desktop: the user will see and use his plain data, but it will be stored on the USB stick and on Dropbox in crypted form.

3.2.10 DNS setup

Ubuntu uses NetworkManager for network related configurations: a useful utility to check what Ubuntu wants to do is

`nm-tool`

which shows the current configuration. Information are kept in

`/etc/NetworkManager/NetworkManager.conf`

and in particular the `DNS` info are the line

`dns=dnsmask`

where `dnsmasq` is a local service which forwards DNS request to external DNS.

To have a "standard" configuration, it is possible to add the usual configuration file `/etc/resolv.conf` with something like:

`nameserver 208.67.222.222`

3.2.11 Remastersys

When the installation is complete, `remastersys` is the tool used to build an ISO of the system. The ISO is limited to 4G in size, for limitation of the tools used by remastersys and the Ubuntu USB booting procedure: for more informations about Remastersys see its website http://www.remastersys.com.

The information about when remastersys is issued are stored in the directory `/home/universusb/.scripts` folder in a file whose name keeps the version of the script and the date in which the script started building the UniversUSB, in the format yymmgghhmmss.

```
DATA=‘date "+%Y%m%d%H%M%S"‘
...
echo UniversUSB_${VERSION}_$DATA >
  /home/$DEFAULT_USERNAME/.scripts/UniversUSB_${VERSION}_$DATA
```

3.2.12 The Setup.sh script

When UniversUSB is used for the first time, it needs some configurations, which are done by a script called `Setup.sh`. This script and some other files are stored in `/home/universusb/.scripts`, and a Desktop icon is prepared.

```
cat > $DESKTOP_DIR/Setup.desktop <<EOF
[Desktop Entry]
Type=Application
Name=Setup
Terminal=true
Exec=/home/$DEFAULT_USERNAME/.scripts/Setup.sh
Icon=/home/$DEFAULT_USERNAME/.scripts/blue-${DEFAULT_USERNAME}.svg
EOF

chmod a+x $DESKTOP_DIR/Setup.desktop
```

The script will be executed by the user the first time that he will use the USB stick and essentially will set up:

1. Dropbox account (an existing one or a new one)
2. encfs on the private-data directory

Chapter 4

References

EncFS
`http://www.arg0.net/encfs` EncFS site, 15

RemasterSys
`http://www.remastersys.com` RemasterSys site, 16

Ubuntu
`http://www.ubuntu.com/` Ubuntu site, 7

UNetBootin
`http://unetbootin.sourceforge.net/` UNetBootin site, 10

UniversUSB
`http://www.universusb.org/` UniversUSB site, 5